RIPE NCC
RIPE NETWORK COORDINATION CENTRE

ASPA
RPKI BGP Path Verification

Tim Bruijnzeels | nog.fi 2026.06| 16 June 2026

What is ASPA?

AS Provider Authorisations (ASPA)

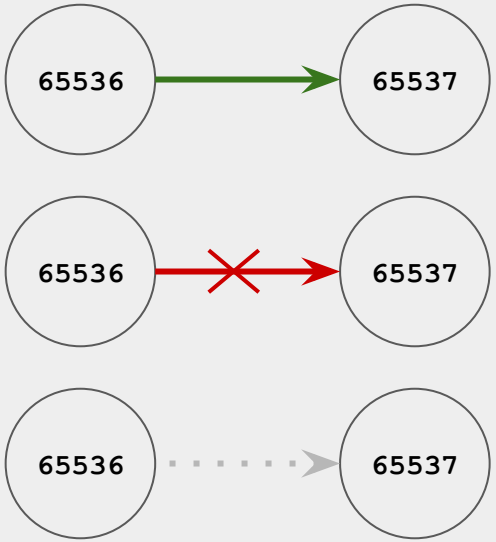
ASPA Object Structure (simplified)

- Like a ROA
- Signed by the holder of one Customer AS
- Authorising one or more Provider ASes

The holder of the Customer AS Number declares which AS Numbers may appear as their Providers in BGP paths

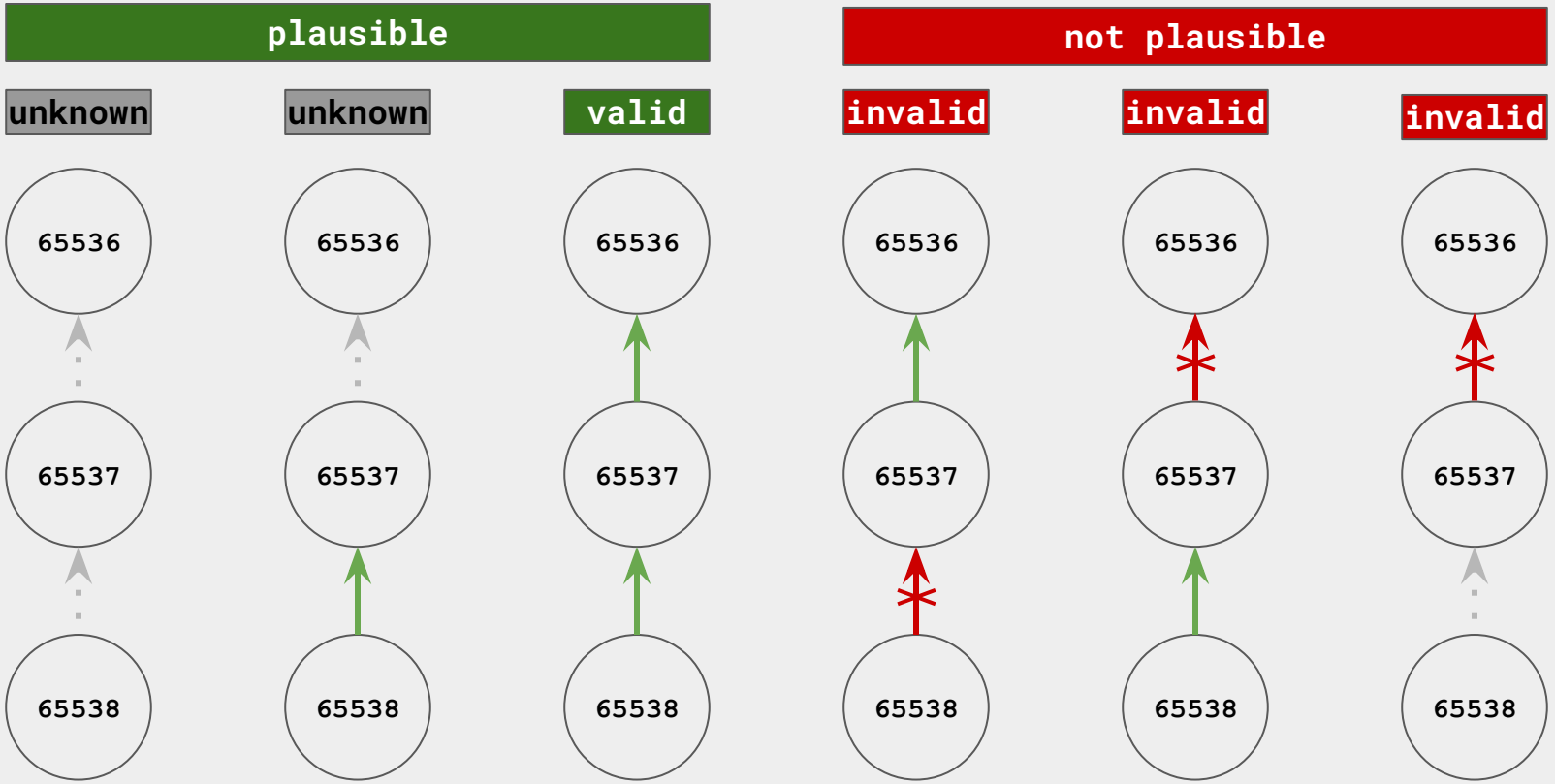
Provider Authorization Function

Fundamental Function to Evaluate AS Pairs



Provider	65536 has an ASPA including 65537
Not Provider	65536 has an ASPA without 65537
No Attestation	65536 has no ASPA

ASPA Verification of Customer Paths



Other Routes? Valley-Free Routing!

Announcements go **up-ramp** from customers to providers to a **single peak** consisting of a shared provider or two adjacent peers and then **down-ramp** from providers to customers.

Announcements that go **up**, then **down** and then **up** (and likely **down**) again have a **valley**.

Valleys are considered harmful route leaks due to:

- Latency
- Congestion
- Cost
- Data security

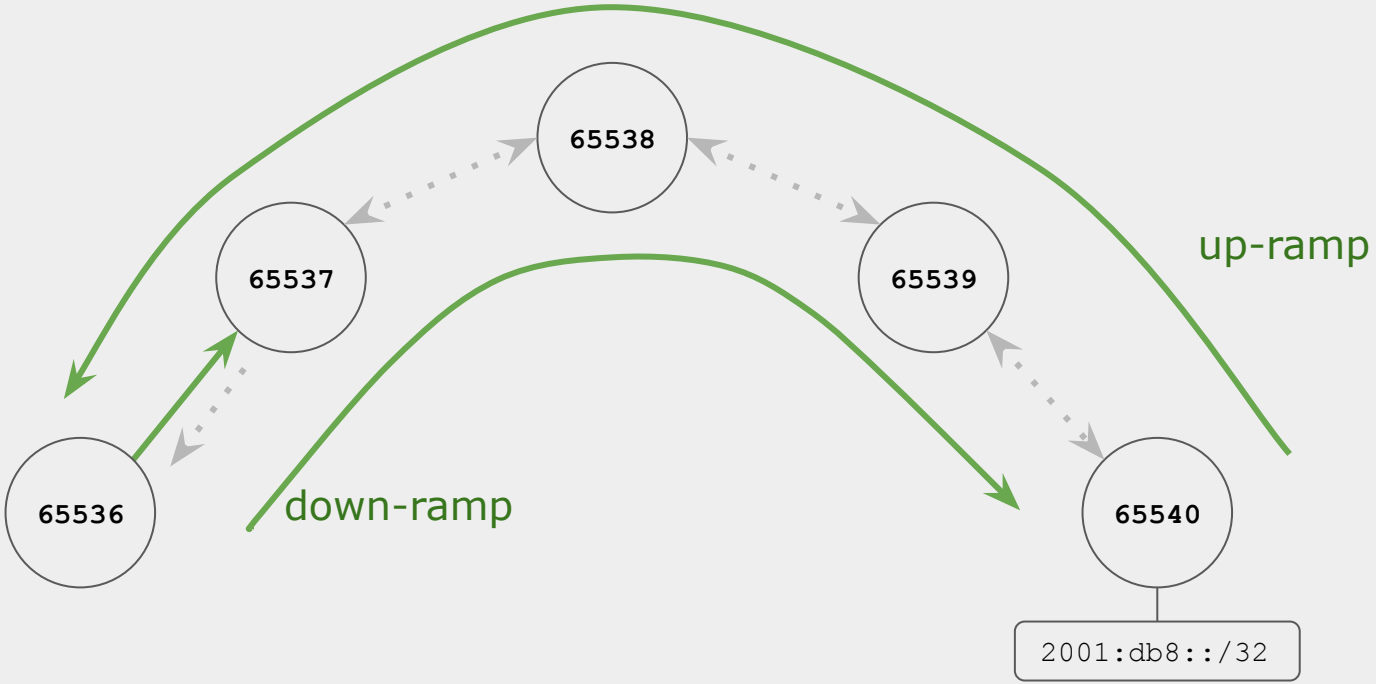
How ASPA Finds Valleys

- Find the…
  - longest **plausible** customer-provider up-ramp from the origin
  - longest **plausible** customer-provider down-ramp from the reverse

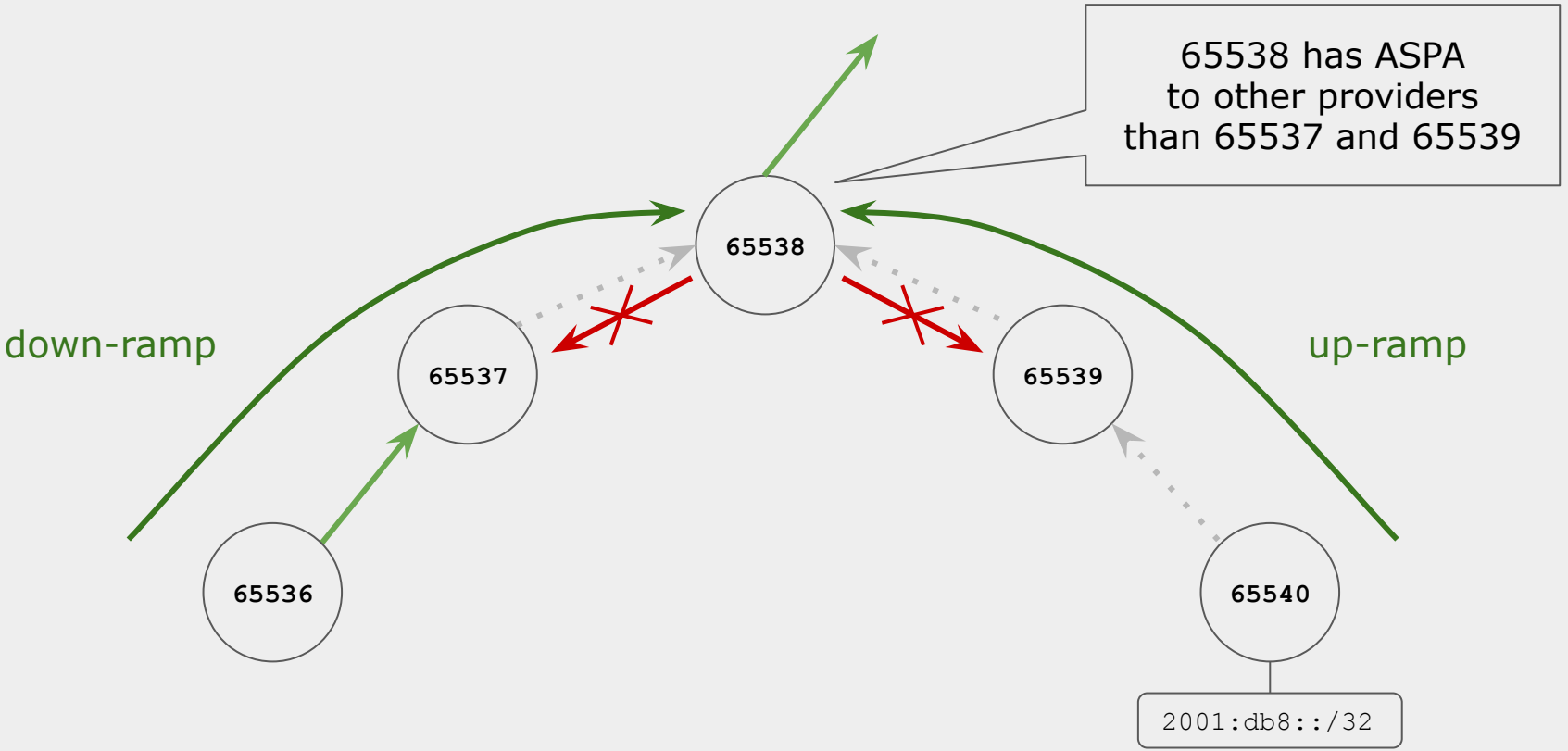
- Okay if…
  - They overlap (e.g. partial deployment)
  - They meet in a single shared provider
  - They meet in a single peering adjacency

- Invalid if they are separated by more adjacencies

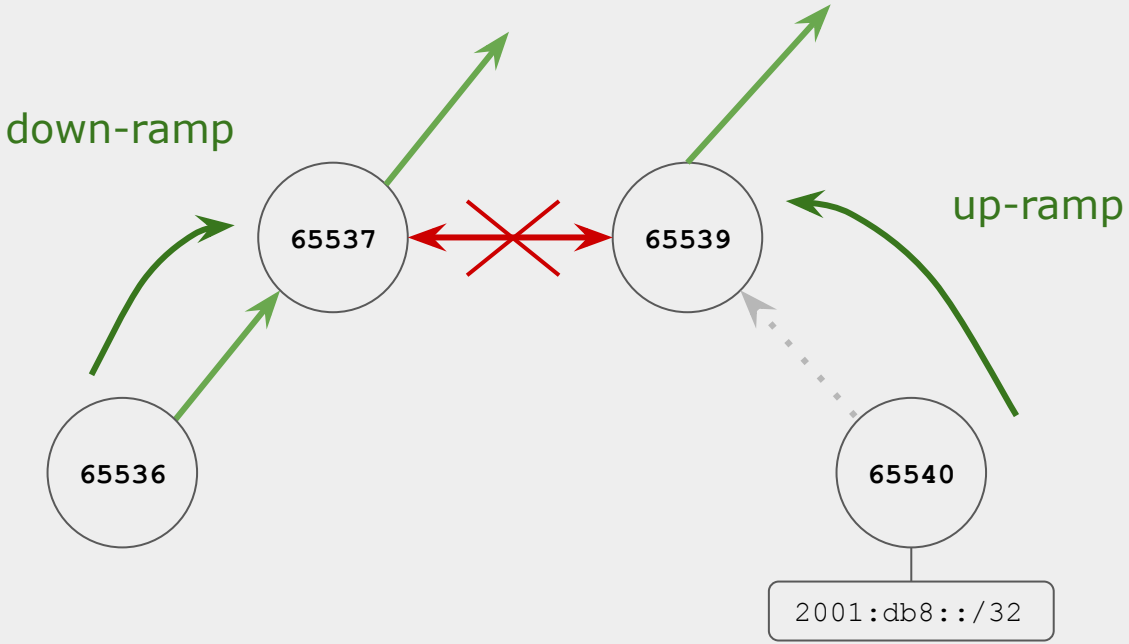
Valley-Free - Overlapping



Valley-Free - Single Provider



Valley-Free - Peering

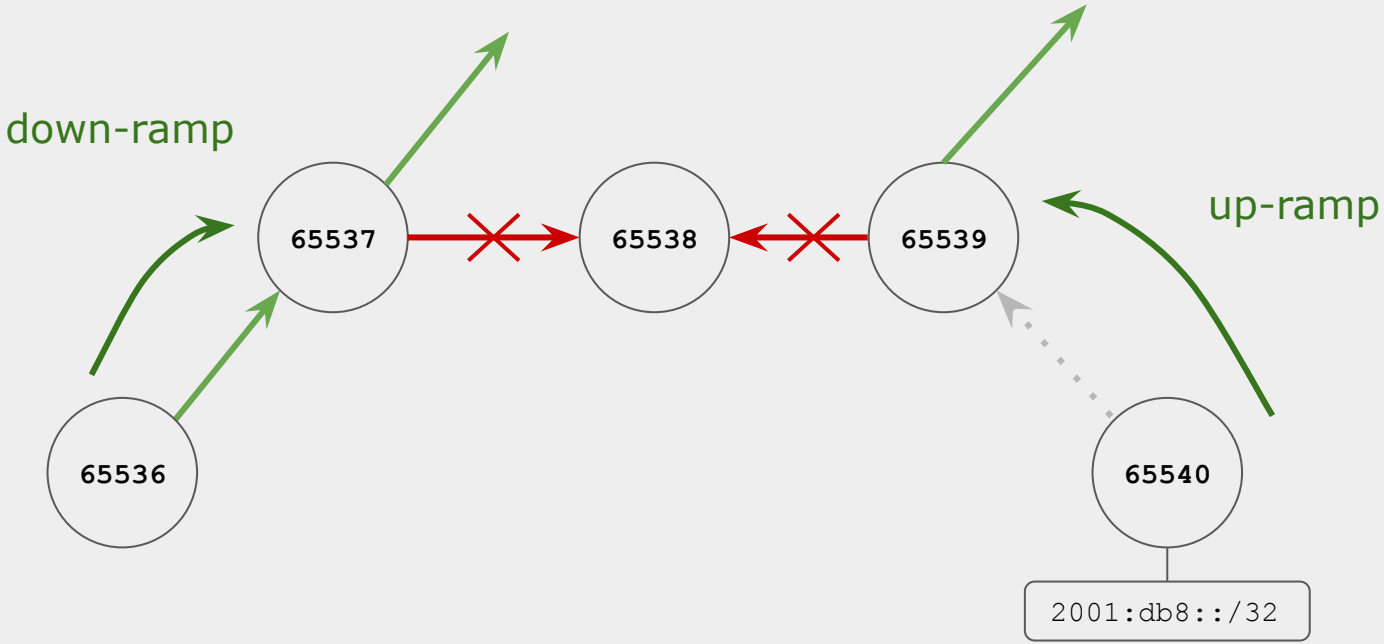


Ramps meet at a single **not-provider** adjacency

Counterintuitive? Perhaps, but… this is a feature.

Otherwise, all peers must be included as providers.

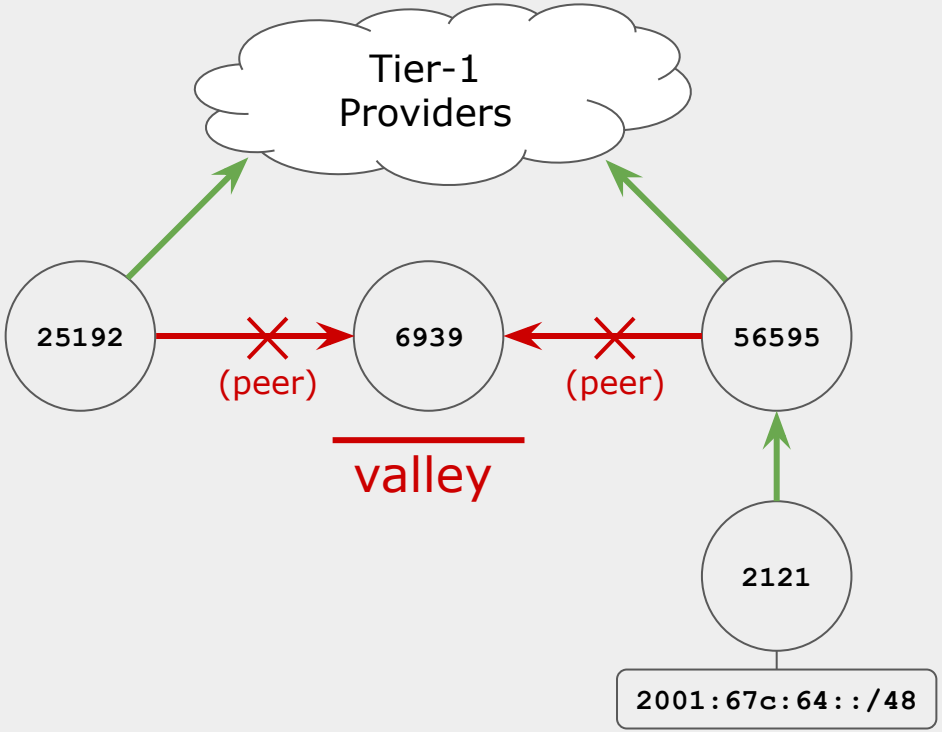
Valley-Free - Valley / Leak



It’s Complicated

(when a peer is not a peer)

RIPE Meeting Validation Case



Meeting network (2121) had one ISP (56595). Both 56595 and 25192 consider 6939 a peer and issued ASPA objects, but 6939 provided IPv6 transit.

ASPA did the **right** thing. Both 56595 and 25192 **told** us that 6939 is not a provider.

Fixes:

➔ 56595 added 6939 to ASPA allowing them to be a provider

➔ Consider using “OTC”: Only-To-Customer (RFC 9234)

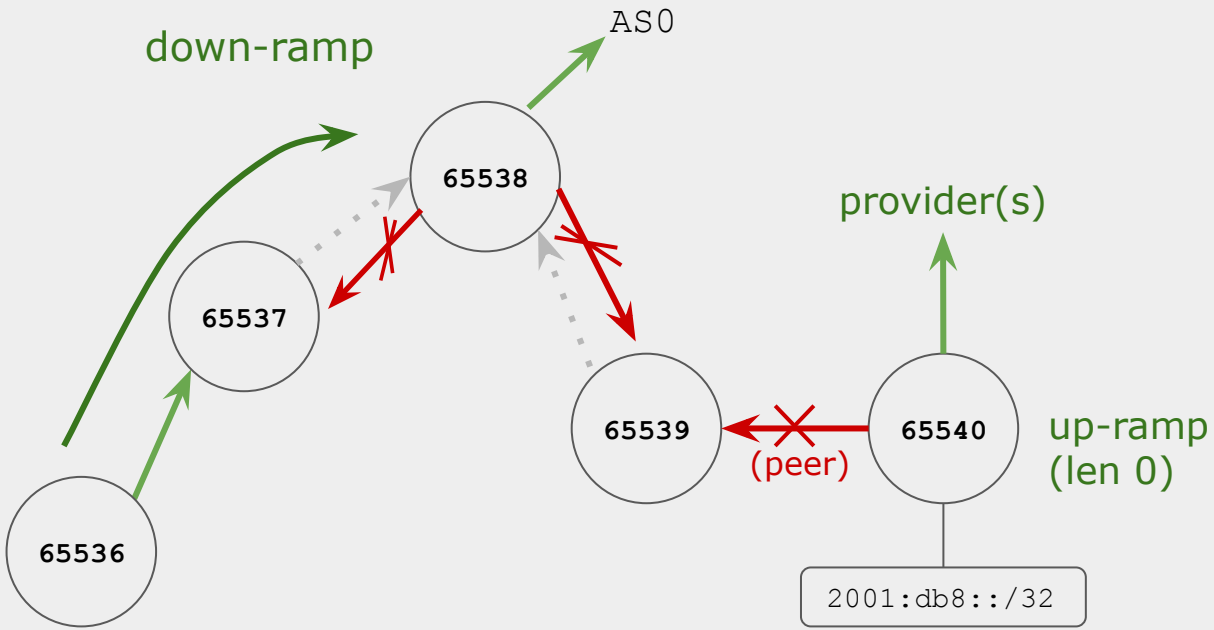
The Curious Case of AS0

The Curious Case of AS0

AS0 is used in ASPA objects to indicate that the **customer AS** has no providers.

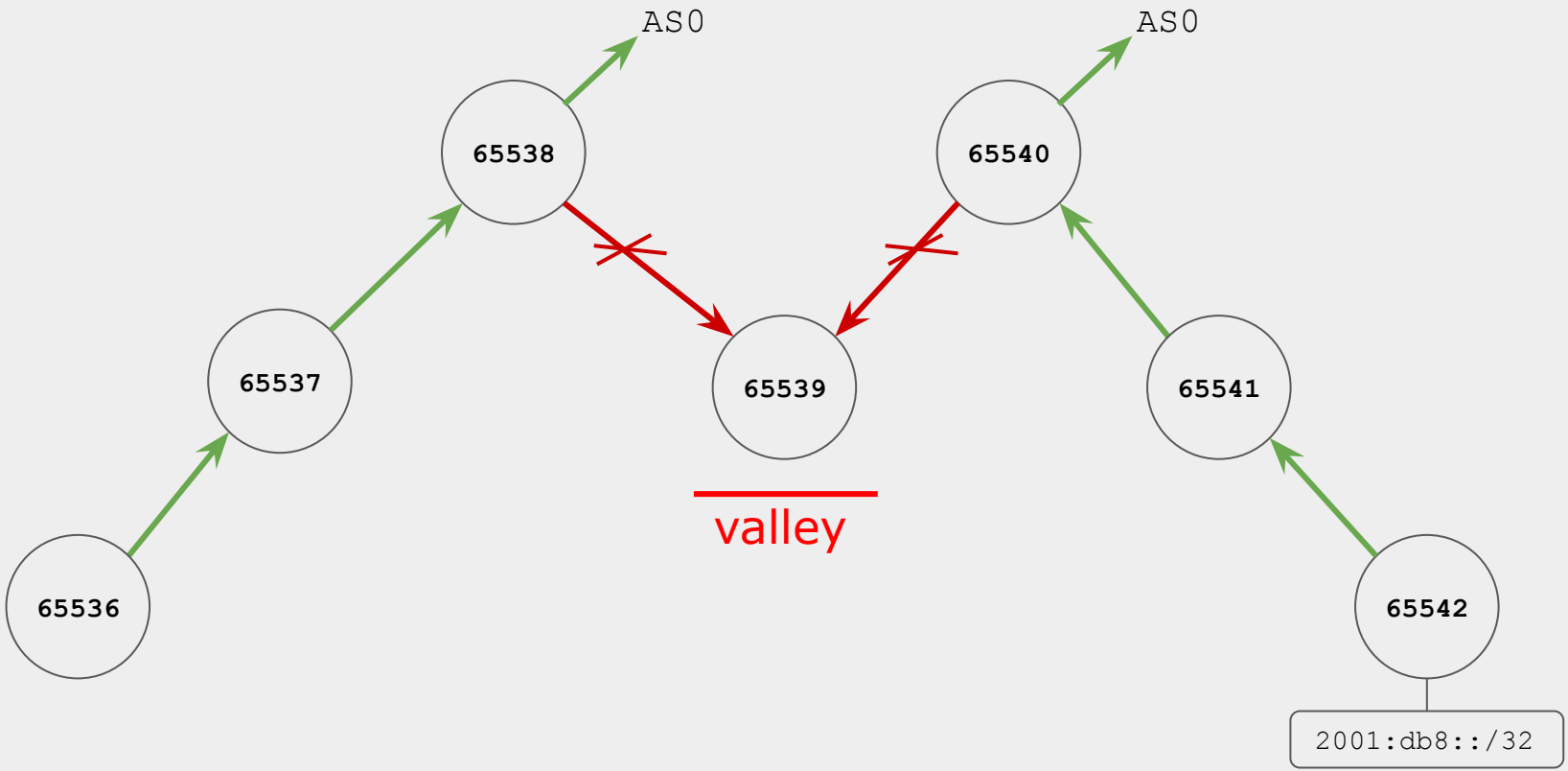
- AS0 is reserved and cannot be on the path
- So every next AS gets “Not Provider”
- Used by
  - Tier-1/Transit-free Networks
  - Transparent Route Servers
  - Route Collectors
  - Or even… unused ASNs

Valley-Free - Leak towards AS0 ASPA



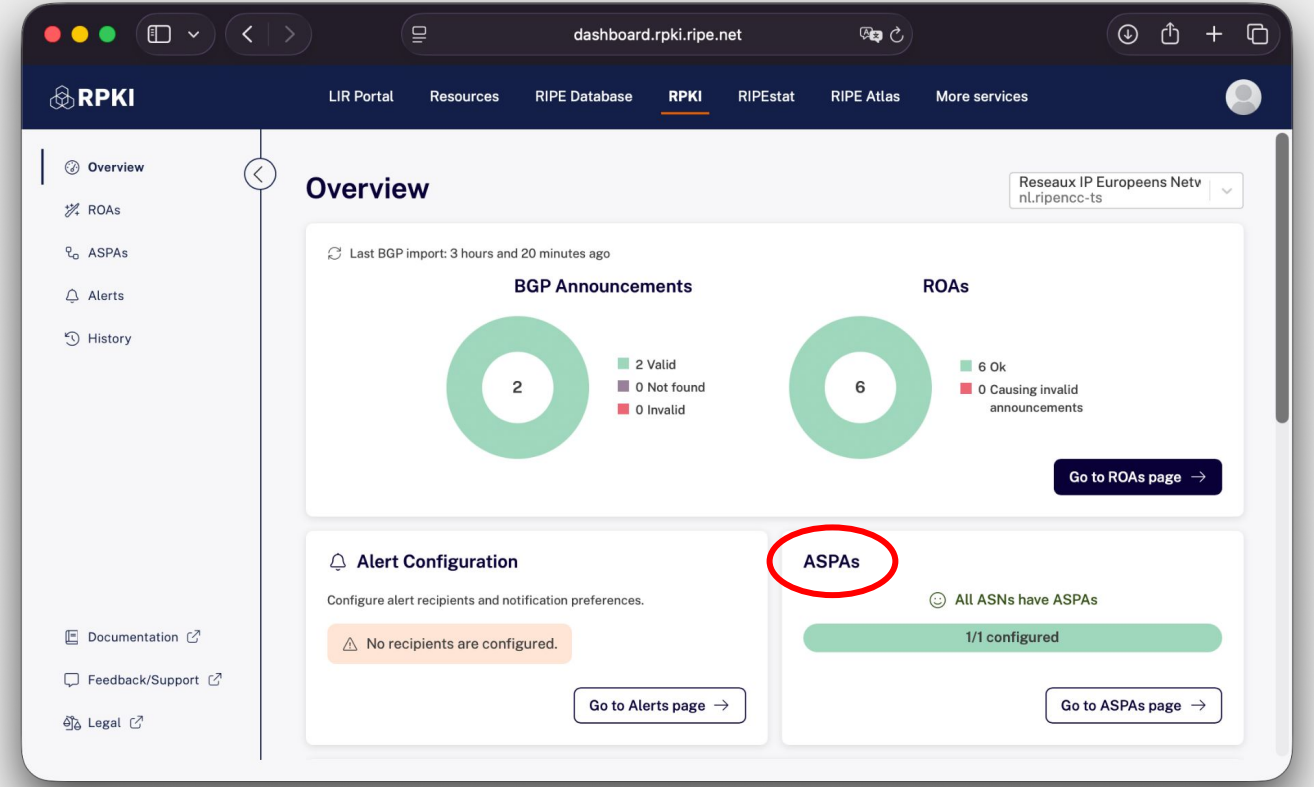
- Tier-1 signing really helps to flag ASPA invalid paths
- Tier-1 validating would stop leaks early

Valley-Free - Leak to second Tier-1

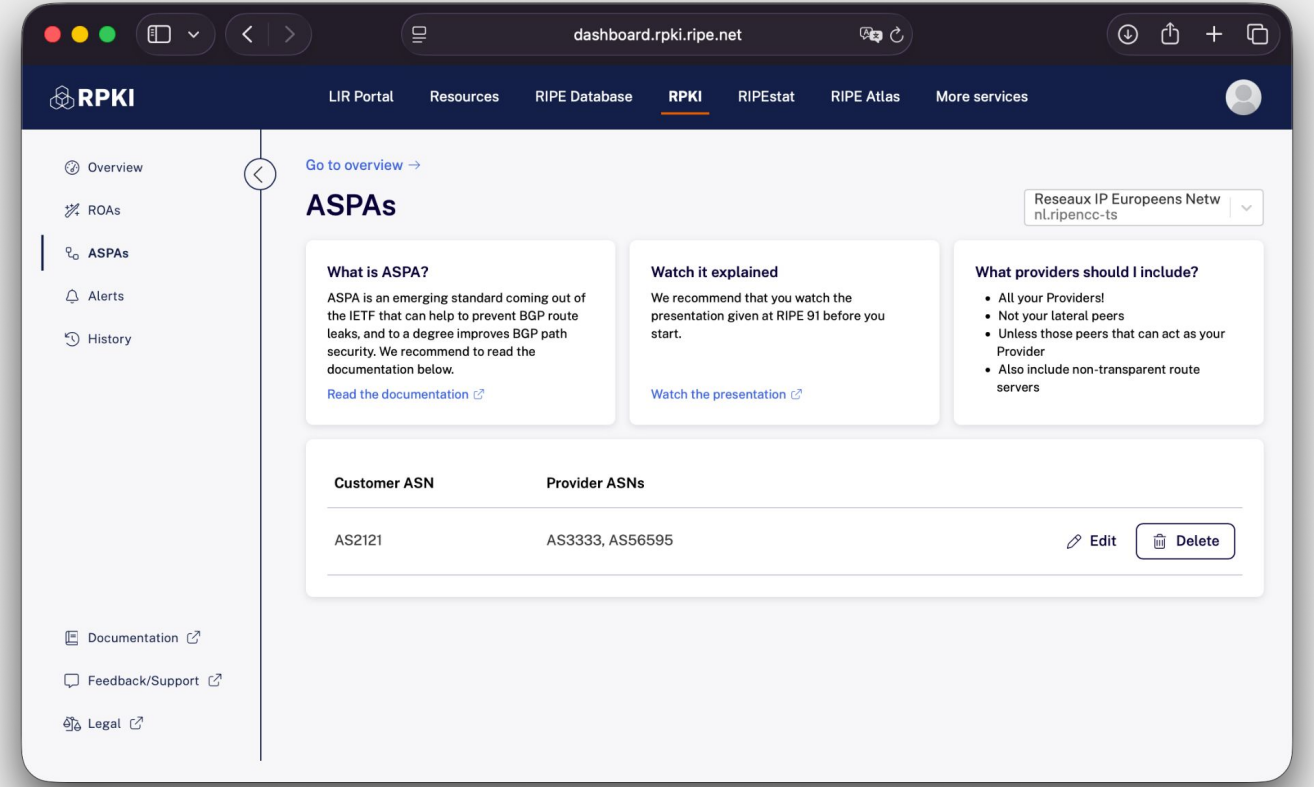


ASPA Signing

RPKI Dashboard



RPKI Dashboard

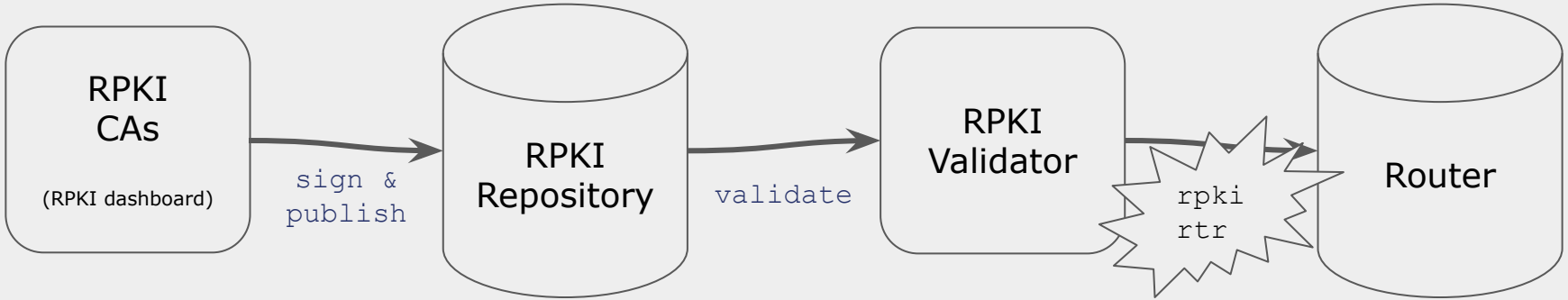


Suggest Providers based on RIS?

- NO!
  - It will include false positives
  - It will not include providers not seen by RIS
- YES!
  - People will forget networks
  - This becomes problematic when people start validating
- The middle?
  - Suggest possible forgotten providers, carefully
  - Avoid click-all-and-save
  - Plan to look at this later this year

ASPA in the Router

From Signing to Router



- ➔ Same deployment model as ROAs
- ➔ No crypto in the router (works on modest hardware)
- ➔ Allows simple config in routers

- ➔ Long standardisation process (yes, even by IETF measures)
  - ◆ object profile, rpki-rtr, path verification
- ➔ Consensus finally reached (it seems)

ASPA Validation Deployment

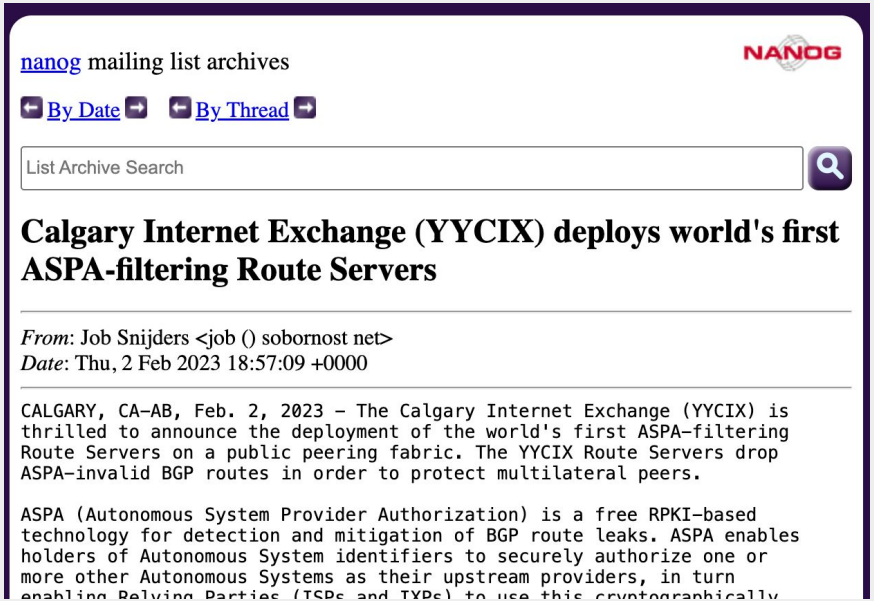
It’s a sharp tool

- As with Route Origin Validation: reject invalid
- Fix your ASPAs:
  - Do NOT sign if you don’t maintain
  - Assume that people will drop invalid

But it’s early days

- Not yet available on many routers, talk to your vendor!
  (Supported on: Bird, OpenBGPd, NIST-BGP-SR. Test phase: Cisco IOS-XR)

YYCIX was first (2023)



nanog mailing list archives

NANOG

By Date By Thread

List Archive Search

Calgary Internet Exchange (YYCIX) deploys world's first ASPA-filtering Route Servers

From: Job Snijders <job () sobornost net>
Date: Thu, 2 Feb 2023 18:57:09 +0000

CALGARY, CA-AB, Feb. 2, 2023 - The Calgary Internet Exchange (YYCIX) is thrilled to announce the deployment of the world's first ASPA-filtering Route Servers on a public peering fabric. The YYCIX Route Servers drop ASPA-invalid BGP routes in order to protect multilateral peers.

ASPA (Autonomous System Provider Authorization) is a free RPKI-based technology for detection and mitigation of BGP route leaks. ASPA enables holders of Autonomous System identifiers to securely authorize one or more other Autonomous Systems as their upstream providers, in turn enabling Relying Parties (ISPs and IXPs) to use this cryptographically

Using:

- rpki-client
- OpenBGPd

(both run on Linux and other systems as well)

Dropping ASPA (and ROV) Invalid in OpenBGPd

```
include "/var/db/rpki-client/openbgpd"

neighbor 5.10.192.4 {
        role customer
        remote-as 51088
}

neighbor 2a02:1668:a2b:1:3::1 {
        role customer
        remote-as 51088
}

# reject ROA invalids
deny quick from ebgp ovs invalid

# reject ASPA invalids
deny quick from ebgp avs invalid
```

https://man.openbsd.org/bgpd.conf

ASPA in the RIPE Meeting Network

Key Configuration of BIRD (3.2.1+)

```
roa4 table roas4;
roa6 table roas6;
aspa table aspas;
attribute enum roa roa_status;
attribute enum aspa aspa_status;
attribute int valid_roa;
attribute int valid_aspa;


protocol rpki rpki_validator {
    roa4 { table roas4; };
    roa6 { table roas6; };
    aspa { table aspas; };
    remote …;
}
```

```
template bgp public_transit_v4 {
    direct;
    local as 2121;
    local role customer;

    ipv4 {
        import filter bgp_in_transit_v4;
        export where proto = "as2121_v4";
        import keep filtered;
        # Needed for automatic RPKI reload
        import table on;
    };
}
```

routinator over rpki-rtr

ASPA in the RIPE Meeting Network

Key Configuration of BIRD (3.2.1+)

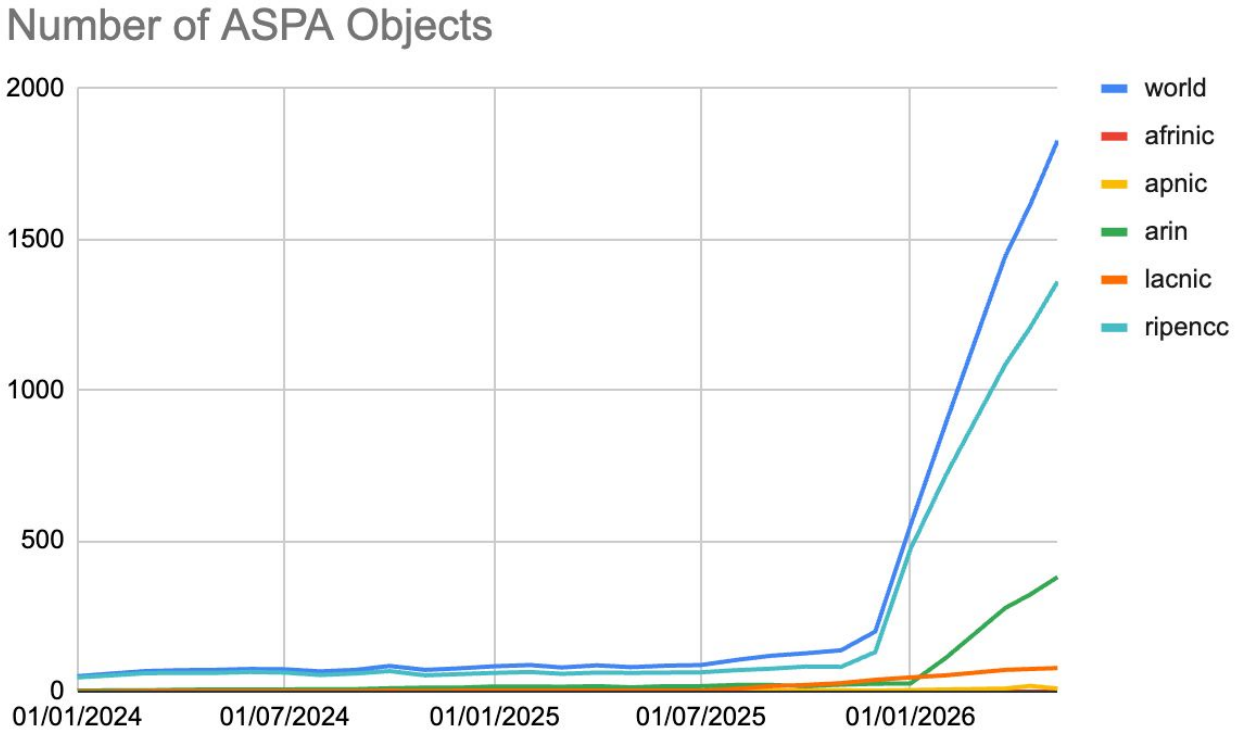
```
filter bgp_in_transit_v4 {
(…)
    roa_status = roa_check(roas4);
    case roa_status {
      ROA_INVALID: {
        valid_roa = 2;
        reject "Invalid ROA (…)";
      }
      ROA_VALID: valid_roa = 1;
      ROA_UNKNOWN: valid_roa = 0;
    }
```

```
    aspa_status = aspa_check_downstream(aspas);
     case aspa_status {
       ASPA_INVALID: {
         valid_aspa = 2;
         reject proto, ": invalid ASPA(…)";
       }
       ASPA_VALID: valid_aspa = 1;
       ASPA_UNKNOWN: valid_aspa = 0;
     }

     accept;
}
```

ASPA Uptake

ASPA Signing Uptake



Before July 2025

- Krill CLI

July 2025

- Krill UI

Nov 2025

- RIPE NCC UI

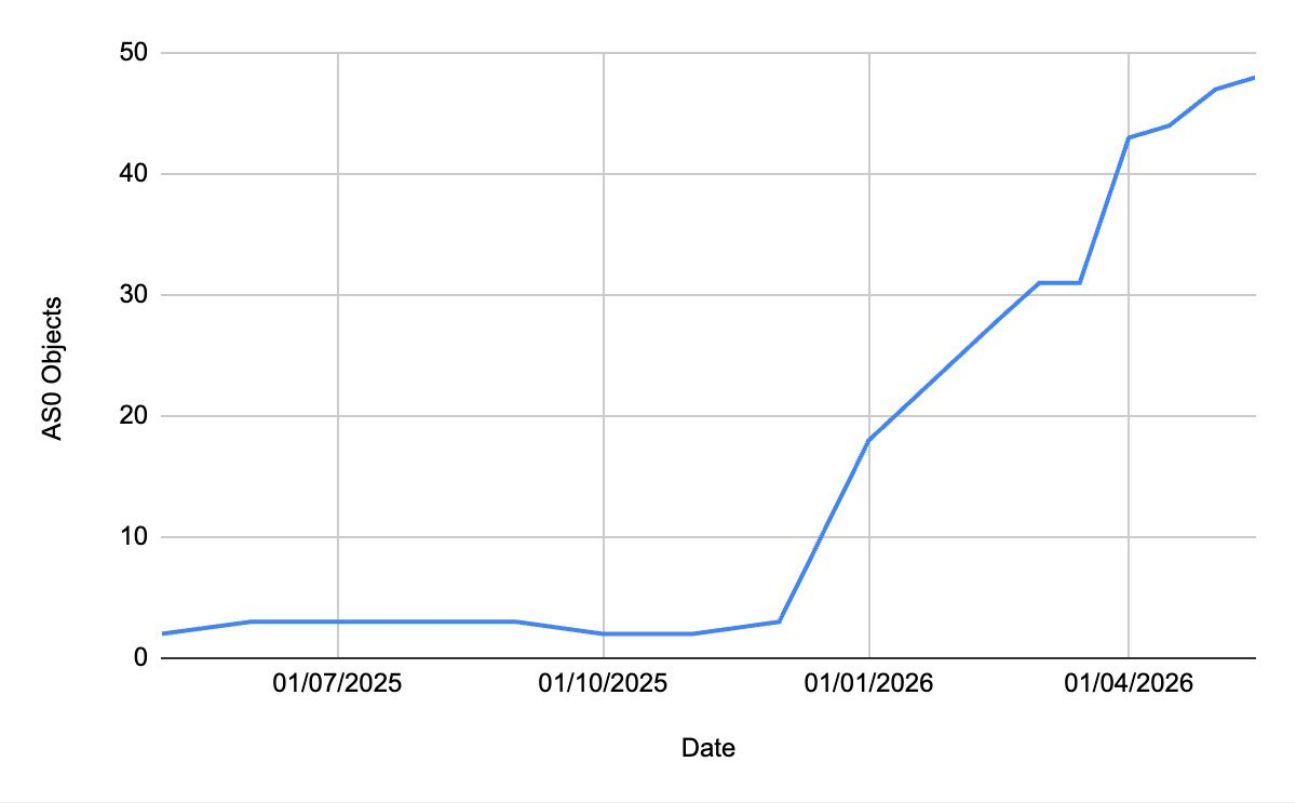
Jan 2026

- ARIN UI

ASPA Validation Uptake

AS0 in the Wild!

AS0 Signing Uptake



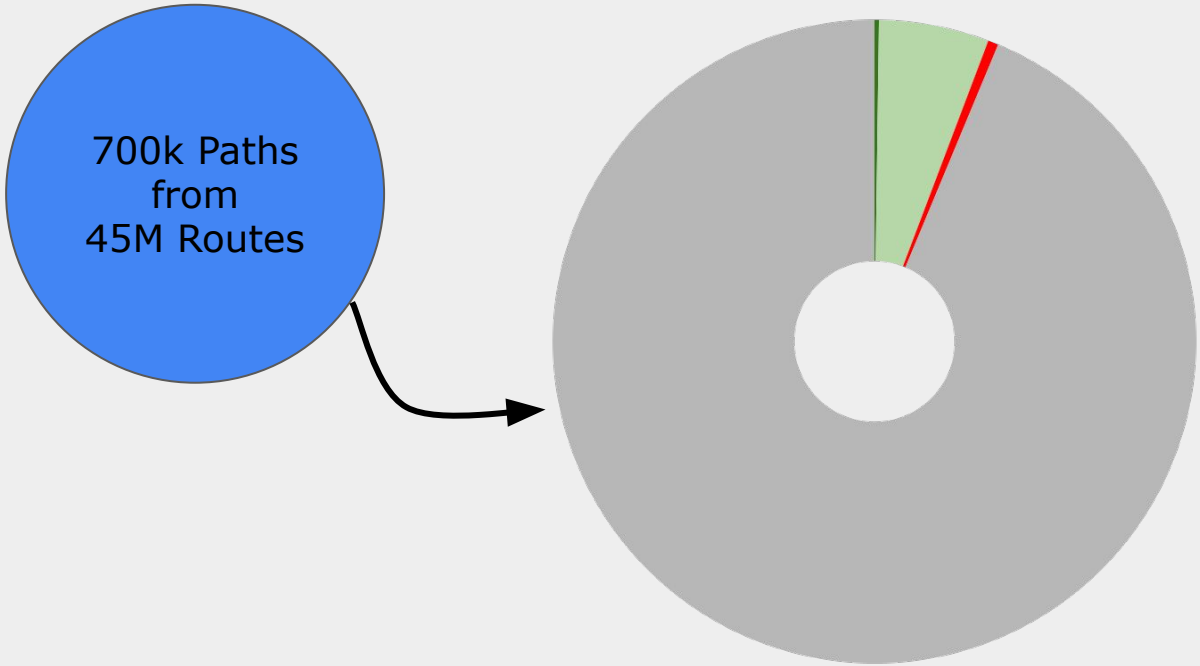
Can We Use AS0 ASPAs to Learn?

AS0 ASPAs signal Provider Free Networks

- Look at **all** routes seen in RIS
- Get the **unique** paths leading up to a **provider free** network
- Remove any duplicated ASes in the path
  (learned this the hard way)

- Then validate these paths, as **customer** paths leading to:
  - Invalid if it contains any ‘not provider’
  - Valid if all hops are to ‘provider’
  - Unknown if partially covered by ‘provider’ and no ‘not provider’
  - Not covered if no other AS has ASPA

What Can We Learn?



Not covered	93.7%
Valid	0.2%
Unknown	5.5%
Invalid	0.5%

Next Steps?

ASPA Suggestions / Analysis

Extend the ASPA UI with suggestions and analysis

- Analysing paths to AS0 ASPA can help
  - We could find all other providers for 56595
  - But not leaks between peers that don’t go through a tier-1
- Evaluate ASPA path validity before publication
  - Show effect on routes for customer AS
  - Leverage known other ASPAs
  - Leverage known role for routes collected in RIS
  - But this still needs a lot of work and thought
- Don’t make it a click and save
  - You *should* know who your transits are
  - You *should* check suggestions
  - And if you know, you can sign ASPAs today

Next?

Dos

- Sign your ASPAs!
  - Maintain your ASPAs
  - Include **all** your transits
- Validate routes
- Reject ASPA **invalid** routes

Don’ts

- Up/down pref (not effective)
- Set communities based on validation state (too much churn)

Questions & Comments

tbruijnzeels@ripe.net